Amendments to the Claims

1.   (currently amended) A method for delivering data within a ~~single~~ data packet comprising the steps of:

disabling a data integrity function of ~~a single~~ said data packet, said data integrity function configured for determining whether data within said ~~single~~ data packet is valid;

calculating data integrity information for each of a plurality of independent data segments to be transmitted within said ~~single~~ data packet; and

transmitting, within said ~~single~~ data packet, said plurality of independent data segments and said data integrity information calculated for each of said plurality of independent data segments.

2.   (original) The method as in claim 1 wherein said data integrity information is a checksum.

3.   (original) The method as in claim 1 wherein said data integrity function is a checksum function.

4.   (currently amended) The method as in claim 1 wherein said ~~single~~ data packet is a ~~single~~ User Datagram Protocol ("UDP") packet.

5.   (original) The method as in claim 1 wherein said independent data segments are GSM-AMR audio frames.

6.   (currently amended) The method as in claim 4 wherein disabling said data integrity function of said ~~single~~ data packet comprises setting a checksum of said ~~single~~ data packet to zero.

7.   (currently amended) The method as in claim 1 further comprising:

receiving said ~~single~~ data packet at a client;

determining whether any of said independent data segments are corrupt based on said data integrity information; and

discarding any independent data segments which are corrupt.

8.   (previously presented) The method as in claim 7 wherein determining whether any of said independent data segments are corrupt comprises:

recalculating said data integrity information for each of said plurality of independent data segments; and

comparing said recalculated data integrity information with said transmitted data integrity information to determine whether any of said independent data segments are corrupt.

9.   (currently amended) An apparatus for delivering data within a ~~single~~ data packet comprising:

a data integrity calculation module for calculating data integrity information for each of a plurality of independent data segments;

a packet generation module for encapsulating, within ~~a single~~ said data packet, said plurality of independent data segments and said data integrity information calculated for each of said plurality of independent data segments and disabling a data integrity function of said ~~single~~ data packet; and

a transmission module for transmitting said ~~single~~ data packet over a network to a destination.

10.   (original) The apparatus as in claim 9 wherein said data integrity information is a checksum.

11.   (original) The apparatus as in claim 9 wherein said data integrity function is a checksum function.

12.   (currently amended) The apparatus as in claim 9 wherein said ~~single~~ data packet is a ~~single~~ User Datagram Protocol ("UDP") packet.

13.   (original) The apparatus as in claim 9 wherein said data segments are GSM-AMR audio frames.

14.   (currently amended) The method as in claim 12 wherein disabling said data integrity function of said ~~single~~ data packet comprises setting a checksum of said ~~single~~ data packet to zero.

15.   (previously presented) A method comprising:

providing a UDP datagram, the UDP datagram having a header and a payload, the payload comprised of a plurality of independent data segments, the header comprising a source port field, a destination port field, a length field, and a datagram checksum;

setting the datagram checksum to zero;

adding a checksum to each independent data segment in the payload; and

sending the modified datagram through to a destination port.

16.   (cancelled)

17.   (currently amended) A machine-readable medium having program code stored thereon which, when executed by a machine, cause said machine to perform the operations of:

disabling a data integrity function of a ~~single~~ data packet, said data integrity function ~~capable of~~ configured for determining whether data within said ~~single~~ data packet is valid;

calculating data integrity information for each of a plurality of independent data segments to be transmitted within said ~~single~~ data packet; and

transmitting, within said ~~single~~ data packet, said plurality of independent data segments and said data integrity information calculated for each of said plurality of independent data segments.

18.   (original) The machine-readable medium as in claim 17 wherein said data integrity information is a checksum.

19.   (original) The machine-readable medium as in claim 17 wherein said data integrity function is a checksum function.

20.   (currently amended) The machine-readable medium as in claim 17 wherein said ~~single~~ data packet is a ~~single~~ User Datagram Protocol ("UDP") packet.

21.   (original) The machine-readable medium as in claim 17 wherein said independent data segments are GSM-AMR audio frames.

22.   (currently amended) The machine-readable medium as in claim 20 wherein disabling said data integrity function of said ~~single~~ data packet comprises setting a checksum of said ~~single~~ data packet to zero.

23.   (currently amended) The machine-readable medium as in claim 17 including program code which causes said machine to perform the additional operations of:

    receiving said ~~single~~ data packet at a client;

    determining whether any of said independent data segments are corrupt based on said data integrity information; and

    discarding any independent data segments which are corrupt.

24.   (previously presented) The machine-readable medium as in claim 23 wherein determining whether any of said independent data segments are corrupt comprises:

recalculating said data integrity information for each of said plurality of

independent data segments; and

comparing said recalculated data integrity information with said transmitted data

integrity information to determine whether any of said independent data segments are

corrupt.